



Inertial parameters identification and joint torques estimation with proximal force/torque sensing

Silvio Traversaro, Andrea del Prete, Serena Ivaldi, Francesco Nori

► To cite this version:

Silvio Traversaro, Andrea del Prete, Serena Ivaldi, Francesco Nori. Inertial parameters identification and joint torques estimation with proximal force/torque sensing. 2015 IEEE International Conference on Robotics and Automation (ICRA 2015), May 2015, Seattle, WA, United States. hal-01131610

HAL Id: hal-01131610

<https://inria.hal.science/hal-01131610>

Submitted on 14 Mar 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Inertial parameters identification and joint torques estimation with proximal force/torque sensing

Silvio Traversaro¹, Andrea Del Prete², Serena Ivaldi³ and Francesco Nori¹

Abstract—Classically robot force control passes through joint torques measurement or estimation. Within this context, classical torque sensing technologies rely on current sensing on motor windings and on torsion sensing on motor shaft. An alternative approach was recently proposed in [1] and combines whole-body distributed 6-axis force/torque (F/T) sensors, gyroscopes, accelerometers and tactile sensors (i.e. artificial skin). A further advantage of this method is that it simultaneously estimates (internal) joint torques and (external) contact forces with no need of joint redesign. As a drawback, the method relies on a model of the robot dynamics, as it consists on reordering the classical recursive Newton-Euler algorithm (RNEA). In this paper we consider the problem of the parametric identification of the robot dynamic model from embedded F/T sensors. We extend recent results on parametric identification [2] by considering an arbitrary reordering of the classical RNEA. The theoretical framework is validated on the iCub humanoid, which is equipped with both 6-axis F/T sensors and joint torque sensors. We estimated the system inertial parameters using only one F/T sensor. We used the obtained parameters to estimate the joint torques (as proposed in [1]) and compared the results with direct joint torque measurements, used in this context only as a ground truth.

I. INTRODUCTION

A fundamental problem in controlling torque-actuated robots is the accurate modeling of their dynamics. Depending on the performed task (e.g. control, simulation, contact detection) we can distinguish two possible approaches [3]: in *structural modeling* the interest is on identifying the real inertial parameters of the robot, while in *predictive modeling* the interest is only on replicating the input-output behavior of the system, the input and output being some measured quantities. In structural modeling the usual approach is to excite the robot with trajectories chosen so as to be optimal for identifying the identifiable (i.e. base) parameters [4]. In predictive modeling, the only concern is to accurately model the input-output response of the dynamical system. This has a significant implication: the interest is not in estimating the “real” parameters, but in getting parameters capable of generalizing predictions across the whole work

space. Interestingly, within this context different regression techniques can be adopted, ranging from parametric [3], semi-parametric [5] and machine learning approaches [6]. A common task that falls within the predictive modeling category is learning inverse dynamics: inputs are positions, velocities and accelerations while outputs are joint torques.

In this paper we tackle a problem that lies in between *structural* and *predictive modeling*. We aim to relax the modeling assumptions in the procedure proposed in [1] to estimate joint torques from embedded 6-axis force/torque (F/T) sensors. This estimation procedure allows us to implement torque control on robots without joint torque sensing. Since most (humanoid) robots are not equipped with joint torque sensors, but have 6-axis F/T sensors, this approach opens the possibility to implement inverse-dynamics control on these “old-generation” robots. Moreover, this is interesting also for new-generation robots, which could be easily equipped with 6-axis F/T sensors, without going through the hassle of redesigning the joints to include torque sensing.

The main drawback of this method is that it relies on the inertial parameters to estimate the joint torques. The goal of this paper is to understand if and to what extent this knowledge is necessary and if we can partially retrieve it through identification procedures similar to the one proposed in [3]. The major technical obstacle lies in the following consideration. We can use F/T measurements to estimate certain inertial parameters (known in literature as *base parameters*): what is the relationship between these parameters and the ones used in [1] to estimate (internal) joint torques and (external) contact forces? In this framework non-parametric techniques have limited appeal and therefore we pursue a parametric approach.

The structure of the paper is as follows. Section II introduces the notation. Section III presents the relationship between the base parameters and the torque estimation scheme presented in [1]; we show that the base parameters are sufficient for estimating internal joint torques and external wrenches from F/T measurements. Section IV proposes an on-line identification scheme to learn the base parameters and uses them to properly estimate joint torques. Section V presents a validation of the proposed methodology on the iCub platform, equipped with both F/T and joint torque sensing, the latter being used only as a ground truth for comparison.

*This paper was supported by the FP7 EU projects CoDyCo (No. 600716 ICT 2011.2.1 Cognitive Systems and Robotics), and Koroibot (No. 611909 ICT-2013.2.1 Cognitive Systems and Robotics).

¹Silvio Traversaro and Francesco Nori are with Robotics, Brain and Cognitive Sciences Department, Istituto Italiano di Tecnologia, Genoa, Italy name.surname@iit.it

²Andrea Del Prete is with the LAAS/CNRS, Toulouse, France adelpret@laas.fr

³Serena Ivaldi is with Inria, Villers-lès-Nancy, F-54600, France; CNRS, Loria, UMR n.7503 and Université de Lorraine, Loria, UMR n.7503, Vandoeuvre-lès-Nancy, F-54500, France. She is also with the Intelligent Autonomous Systems Lab., TU Darmstadt, Germany. serena.ivaldi@inria.fr

II. NOTATION

In this paper we use the spatial vector algebra¹ as defined in [8]. However, we extend the notation used in [8] to define the concepts of topology, serialization and base link. While the topology is an intrinsic property of the structure, the serialization and the base link are arbitrary choices that affect computations. The serialization defines an order for links and joints. The base link instead induces a spanning tree that defines the directions in which forces and accelerations are propagated for computations. In the following, to simplify the notation we consider only properly connected trees with one-degree-of-freedom joints.

1) *Topology*: a robot is composed by N_B links, which constitute the link set L , and $N_J = N_B - 1$ joints, represented as unordered pairs of links contained in the joint set J .

2) *Serialization*: a serialization is a sequence of links and a sequence of joints, which we define as two bijections²:

$$\begin{aligned}\mathcal{L} &: 0, 1, \dots, N_B - 1 \rightarrow L \\ \mathcal{J} &: 1, 2, \dots, N_J \rightarrow J\end{aligned}$$

3) *Spanning tree*: a spanning tree is uniquely defined by its root $b \in L$, which we call “base link” of the spanning tree. Given a spanning tree, any link $l \in L$ has a unique parent $\lambda_b(l)$, and by abuse of notation, any joint $j \in J$ has a unique parent link $\lambda_b(j)$ and child link $\mu_b(j)$. We also define $\nu_b(j)$ as the set of the links belonging to the subtree starting at joint j .

4) *Spatial vectors*: as in [8] we denote by M^6 the spatial motion vector space, i.e. the space of linear and angular velocities. Sometimes it will be useful to distinguish between the linear and the angular components of a vector $\mathbf{v} \in M^6$ by denoting $\mathbf{v} = [\boldsymbol{\omega}_l \quad \bar{\mathbf{v}}_l]$. Similarly, an acceleration \mathbf{a} is decomposed in its linear and angular components as follows $[\dot{\boldsymbol{\omega}}_l \quad \bar{\mathbf{a}}_l]$. As in [8] we denote by F^6 the spatial force vector space, i.e. the space of force and torques.

5) *Vector transformations*: the front superscript indicates the frame of reference in which a spatial vector is expressed, e.g. $^a\mathbf{f}$ is the spatial force vector $\mathbf{f} \in F^6$ expressed in the frame of reference a . To simplify the notation, superscripts are omitted when referred to the reference frame associated to link on which the force is acting (i.e. the spatial force vector $\mathbf{f}_l \in F^6$ acting on link l is indicated \mathbf{f}_l instead of $^l\mathbf{f}_l$ when expressed in the link l reference frame. With $^b\mathbf{X}_a^*$ we denote the matrix that transforms a spatial force vector from the reference frame a to the reference frame b , i.e. $^b\mathbf{f} = ^b\mathbf{X}_a^* \mathbf{f}$.

6) *Symmetric matrix serialization*: in the paper we will make use of the vech operator [9], which turns a symmetrical matrix into a vector:

$$\text{vech} \left(\begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \right) = [I_{xx} \ I_{xy} \ I_{xz} \ I_{yy} \ I_{yz} \ I_{zz}]^\top. \quad (1)$$

¹A brief tutorial is available in [7].

²We chose to use two different numbering styles for links and joints to follow [8].

The operator $\boldsymbol{\omega} \bullet$ is defined such that $\boldsymbol{\omega} \bullet \text{vech}(\mathbf{I}) = \mathbf{I}\boldsymbol{\omega}$:

$$\boldsymbol{\omega} \bullet = \begin{bmatrix} \omega_x & \omega_y & \omega_z & 0 & 0 & 0 \\ 0 & \omega_x & 0 & \omega_y & \omega_z & 0 \\ 0 & 0 & \omega_x & 0 & \omega_y & \omega_z \end{bmatrix} \quad (2)$$

A. Floating base dynamics

We here consider the formulation of floating base dynamics as presented in [8]. We slightly modify the notation by including in the front subscripts a \mathcal{J} and/or a b whenever a certain quantity depends on the choice of the joint serialization and/or on the choice of the base link. The subscript, as usual, is used to represent the fact that a certain quantity is associated with a link $l \in L$ (including the base b) or with a joint $j \in J$.

$$\begin{bmatrix} {}^b\mathbf{I}^c \\ {}_{b,\mathcal{J}}\mathbf{F}_b^\top \end{bmatrix} \begin{bmatrix} {}^b\mathbf{a}_b \\ {}_{b,\mathcal{J}}\mathbf{H} \end{bmatrix} \begin{bmatrix} \mathbf{a}_b \\ \ddot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} {}^b\mathbf{p}^c \\ {}_{b,\mathcal{J}}\mathbf{C} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_b \\ \boldsymbol{\tau} \end{bmatrix} + \sum_{l \in L} \begin{bmatrix} {}^b\mathbf{X}_l^* \\ {}_{b,\mathcal{J}}\mathbf{J}_l^\top \end{bmatrix} \mathbf{f}_l^x \quad (3)$$

where:

\mathbf{a}_b	spatial acceleration of the floating base
\mathbf{f}_b	spatial force at floating base
\mathbf{q}	joint positions, depends on \mathcal{J} but front subscript is omitted
${}^b\mathbf{I}^c$	composite rigid body inertia of the tree
${}^b\mathbf{p}^c$	spatial bias force of the composite tree
$\boldsymbol{\tau}$	joint torques, depends on \mathcal{J} but front subscript is omitted
\mathbf{f}_l^x	the external wrench acting on link l
${}_{b,\mathcal{J}}\mathbf{F}_b$	joint Jacobian of the tree spatial momentum
${}_{b,\mathcal{J}}\mathbf{H}$	joint inertia matrix
${}_{b,\mathcal{J}}\mathbf{C}$	the bias torques considering Coriolis, centrifugal and gravitational effects
${}_{b,\mathcal{J}}\mathbf{J}_l$	the Jacobian for link l

In classical floating base representations the spatial force at the base is assumed to be equal to zero, $\mathbf{f}_b = 0$, but we drop this assumption in the computations that will follow in the next sections.

III. EXTERNAL CONTACT WRENCH AND INTERNAL TORQUE ESTIMATION

In this section we review the approach proposed in [1] to simultaneously estimate (internal) torques and (external) contact wrenches by exploiting embedded whole-body distributed F/T sensors, accelerometers, gyroscopes and tactile sensors. As represented in Fig. 1, the procedure consists in cutting the floating-base tree at the level of the (embedded) F/T sensors obtaining multiple subtrees. Each subtree can be considered as an independent articulated rigid body chain subject to the differential equation (3). Among the multiple forces acting on the subtree, we can distinguish those measured by the F/T sensors (green arrows in Fig. 1) and other unknown forces (red arrows) typically due to external contacts. By assuming one unknown force per subtree we can estimate it together with the joint torques. The procedure originally described in [1] consists in sensing the contact

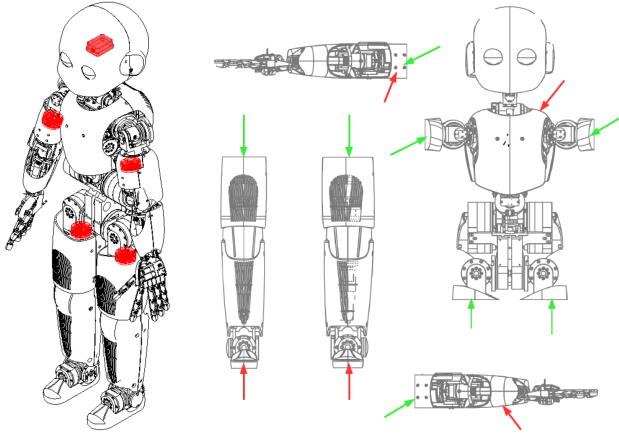


Fig. 1. The left picture shows the position of F/T sensors on the iCub humanoid. The right picture shows the induced iCub partitioning. Each obtained subpart can be considered an independent floating-base structure subject to an external wrench which coincides with the one measured by the F/T sensor (green arrow). Red arrows represent possible location for the unknown external wrenches.

with the artificial skin and using this information to define the base β on the contact link. With this assumption (3) can be rearranged as follows:

$$\begin{bmatrix} \mathbf{f}_\beta \\ \boldsymbol{\tau} \end{bmatrix} = \sum_{l \in \mathcal{L}} \begin{bmatrix} {}^\beta \mathbf{X}_l^* \\ {}_{\beta, \mathcal{J}} \mathbf{J}_l^\top \end{bmatrix} \mathbf{f}_l^x - \begin{bmatrix} {}^\beta \mathbf{I}^c & {}_{\beta, \mathcal{J}} \mathbf{F}_\beta \\ {}_{\beta, \mathcal{J}} \mathbf{F}_\beta^\top & {}_{\beta, \mathcal{J}} \mathbf{H} \end{bmatrix} \begin{bmatrix} \mathbf{a}_\beta \\ \ddot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} {}_\beta \mathbf{p}^c \\ {}_{\beta, \mathcal{J}} \mathbf{C} \end{bmatrix}$$

and computations efficiently performed with a classical recursive Newton-Euler algorithm rearranged according to the new base, therefore the name reordered recursive Newton-Euler algorithm (rRNEA). Position, velocities and accelerations can be either measured or estimated from the available accelerometers, gyroscopes and joint encoders.

A. Estimation of floating base dynamics

It is well known [10] that the right-hand side of (3) can be rearranged linearly with respect to a vector of dynamic parameters ϕ . In the following, we use the front subscripts \mathcal{L} and/or b to denote quantities that depend on the choice of the link serialization and/or on the choice of the base link. Replacing the right-hand side of equation (3) with its linear parametrization we have:

$$\begin{bmatrix} {}_{b, \mathcal{L}} \mathbf{Y}_n \\ {}_{b, \mathcal{L}} \mathbf{Y}_c \end{bmatrix} \mathcal{L} \phi = \begin{bmatrix} \mathbf{f}_b \\ \boldsymbol{\tau} \end{bmatrix} + \sum_{l \in \mathcal{L}} \begin{bmatrix} {}^b \mathbf{X}_l^* \\ \mathbf{J}_l^\top \end{bmatrix} \mathbf{f}_l^x, \quad (4)$$

with

$$\phi^{\mathcal{L}} = \begin{bmatrix} \phi_{\mathcal{L}(0)}^\top & \phi_{\mathcal{L}(1)}^\top & \dots & \phi_{\mathcal{L}(N_B-1)}^\top \end{bmatrix}^\top \in \mathbb{R}^{10N_B},$$

$$\phi_l^\top = [m_l \quad m_l \mathbf{c}_l^\top \quad \text{vech}(\bar{\mathbf{I}}_{Ol})^\top]^\top \in \mathbb{R}^{10},$$

and:

m_l	mass of link l
$m_l \mathbf{c}_l$	first moment of mass of link l
$\bar{\mathbf{I}}_{Ol}$	3D rotational inertia matrix of link l

Remark: in literature, the inertia matrix is often expressed w.r.t. the center of mass of the link (written as $\bar{\mathbf{I}}_C$ [8]). However, we express it w.r.t. a different reference frame - fixed to link l but not located at the center of mass - because otherwise the dynamic equations would not be linear in the inertial parameters.

1) *Identifiable subspaces:* The parametric representation of the robot dynamics (4) has been widely used to get an estimation of the dynamic parameters ϕ from available measurements, denoted hereafter \mathbf{y} . Typical approaches assume null external forces ($\mathbf{f}_l^x = 0$) and exploit either torque [3] ($\mathbf{y} = \boldsymbol{\tau}$) or base force [2] ($\mathbf{y} = \mathbf{f}_b$) measurements. With these assumptions, rows in (4) can be rearranged as $\mathbf{Y}\phi = \mathbf{y}$ and an estimation of ϕ can be obtained by considering repeated measurements $\mathbf{y}^1, \dots, \mathbf{y}^N$ and the associated values of the regression matrix $\mathbf{Y}^1, \dots, \mathbf{Y}^N$, related as follows:

$$\begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{Y}^2 \\ \vdots \\ \mathbf{Y}^N \end{bmatrix} \phi = \begin{bmatrix} \mathbf{y}^1 \\ \mathbf{y}^2 \\ \vdots \\ \mathbf{y}^N \end{bmatrix}. \quad (5)$$

In practice, it is well known that the matrix that multiplies ϕ is rank deficient regardless of the number of measured samples N . More specifically, the following null space can be defined:

$$N_{\mathbf{Y}} = \{ \phi \in \mathbb{R}^{10N_B} : \mathbf{Y}({}^b \mathbf{R}_w, \mathbf{v}_b, \mathbf{a}_b, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \phi = 0, \\ \forall {}^b \mathbf{R}_w \in SO(3), \mathbf{v}_b, \mathbf{a}_b \in \mathbb{M}^6, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^{N_J} \}. \quad (6)$$

In general the space $N_{\mathbf{Y}}$ is non-empty as a consequence of fact that the columns of \mathbf{Y} are linearly dependent for any choice of the robot position, velocity and acceleration. Only certain linear combinations of the elements of ϕ influence the measurements and these combinations can be obtained as $\phi = \mathbf{B}\pi$, being \mathbf{B} a matrix whose columns are an orthonormal base of the so called identifiable subspace $I_{\mathbf{Y}} = N_{\mathbf{Y}}^\perp$. It is then possible to reformulate (5) as:

$$\begin{bmatrix} \mathbf{Y}^1 \mathbf{B} \\ \mathbf{Y}^2 \mathbf{B} \\ \vdots \\ \mathbf{Y}^N \mathbf{B} \end{bmatrix} \pi = \begin{bmatrix} \mathbf{y}^1 \\ \mathbf{y}^2 \\ \vdots \\ \mathbf{y}^N \end{bmatrix} \rightarrow \mathbf{G}_N \pi = \mathbf{g}_N, \quad (7)$$

with obvious definition for the matrix \mathbf{G}_N and the vector \mathbf{g}_N . Classically, equation (7) has been used for the estimation of the base parameters associated with a certain measurement \mathbf{y} . More recently, it has been questioned whether the estimated base parameters can be used to predict other dynamic quantities different from the measurement \mathbf{y} itself. Within this context, in [2] it was shown that base parameters associated with forces at base ($\mathbf{y} = \mathbf{f}_b$) can be used to predict joint torques $\boldsymbol{\tau}$. The result is obtained by showing that the identifiable subspace associated to ${}_{b, \mathcal{L}} \mathbf{Y}_c$ is a subset of the one associated with ${}_{b, \mathcal{L}} \mathbf{Y}_n$. In the present paper we consider the problem of understanding if the base parameters associated with forces at base ($\mathbf{y} = \mathbf{f}_b$) can be used to implement the rRNEA described in Section III. Since the procedure consists in redefining the base b at the contact

link, all we have to understand is the relationship between ${}_{b,\mathcal{L}}\mathbf{Y}_n$ and ${}_{\beta,\mathcal{L}}\mathbf{Y}_n$ for arbitrary choices of the base β .

B. Base parameterization of the regressor structure

In this section we discuss the structure of the matrices ${}_{b,\mathcal{L}}\mathbf{Y}_n$ and ${}_{b,\mathcal{L}}\mathbf{Y}_c$ in detail. With respect to previous literature we explicitly take into account how the choice of b , \mathcal{L} and \mathcal{J} influences the matrix structure. The reason for this new formulation lies in the fact that in the following sections we will try to understand how the identifiable subspaces associated to ${}_{b,\mathcal{L}}\mathbf{Y}_n$ and ${}_{b,\mathcal{L}}\mathbf{Y}_c$ change with the choice of the base b . First, we consider the dynamic equation of the generic link $l \in L$:

$${}^l\mathbf{f}_l^B = \mathbf{I}_l \mathbf{a}_l + \mathbf{v}_l \times^* \mathbf{I}_l \mathbf{v}_l = {}^l\mathbf{A}_l^B \phi_l, \quad (8)$$

having defined:

$${}^l\mathbf{A}_l^B = \begin{bmatrix} 0 & -(\bar{\mathbf{a}}_l + \boldsymbol{\omega}_l \times \bar{\mathbf{v}}_l) \times & \dot{\boldsymbol{\omega}}_l \bullet + \boldsymbol{\omega}_l \times \boldsymbol{\omega}_l \bullet \\ \bar{\mathbf{a}}_l + \boldsymbol{\omega}_l \times \bar{\mathbf{v}}_l & \dot{\boldsymbol{\omega}}_l \times + (\boldsymbol{\omega}_l \times)(\boldsymbol{\omega}_l \times) & 0 \end{bmatrix}$$

and:

\mathbf{f}_l^B	net spatial force acting on body l
\mathbf{I}_l	spatial inertia of body l
\mathbf{a}_l	spatial acceleration of body l
\mathbf{v}_l	spatial velocity of body l

In equation (8) we have not explicitly indicated the gravitational spatial force ${}^l\mathbf{f}_l^g$ associated with the gravitational acceleration \mathbf{a}_g . Its contribution on the link $l \in L$ can be expressed as follows:

$${}^l\mathbf{f}_l^g = \mathbf{I}_l \mathbf{a}_g = {}^l\mathbf{A}_g \phi_l, \quad (9)$$

where:

$${}^l\mathbf{A}_g = \begin{bmatrix} 0 & -{}^l\bar{\mathbf{a}}_g \times & 0 \\ {}^l\bar{\mathbf{a}}_g & 0 & 0 \end{bmatrix}.$$

The net spatial force on link $l \in L$ including gravity is therefore obtained by summing (8) and (9):

$${}^l\mathbf{f}_l = {}^l\mathbf{f}_l^B + {}^l\mathbf{f}_l^g = {}^l\mathbf{A}_l^B \phi_l + {}^l\mathbf{A}_g \phi_l = \mathbf{A}_l \phi_l \quad (10)$$

The explicit expression for ${}_{b,\mathcal{L}}\mathbf{Y}_n$ can be derived as in [11] and takes the following form:

$${}_{b,\mathcal{L}}\mathbf{Y}_n = \begin{bmatrix} {}^b\mathbf{X}_{\mathcal{L}(0)}^* \mathbf{A}_{\mathcal{L}(0)} {}^b\mathbf{X}_{\mathcal{L}(1)}^* \mathbf{A}_{\mathcal{L}(1)} \cdots \\ \cdots {}^b\mathbf{X}_{\mathcal{L}(N_B-1)}^* \mathbf{A}_{\mathcal{L}(N_B-1)} \end{bmatrix}. \quad (11)$$

Given the joint serialization \mathcal{J} , the generic joint $j \in J$ is associated with a certain number of rows of the matrix ${}_{b,\mathcal{L}}\mathbf{Y}_c$. The number of rows is determined by the number of degrees of freedom. The rows associated to $j \in J$ are denoted by ${}_{b,\mathcal{L}}\mathbf{Y}_c^j$ and can be computed as follows:

$${}_{b,\mathcal{L}}\mathbf{Y}_c^j = {}^{\mu_b(j)}\mathbf{S}_{\lambda_b(j),\mu_b(j)}^\top \left[\sigma_b^{\mathcal{L}}(j, \mathcal{L}(0))^{\mu_b(j)} \mathbf{X}_{\mathcal{L}(0)}^* \mathbf{A}_{\mathcal{L}(0)} \cdots \right. \\ \left. \cdots \sigma_b^{\mathcal{L}}(j, \mathcal{L}(N_B-1))^{\mu_b(j)} \mathbf{X}_{\mathcal{L}(N_B-1)}^* \mathbf{A}_{\mathcal{L}(N_B-1)} \right], \quad (12)$$

where $\sigma_b^{\mathcal{L}}(j, i) = 1$ if $\mathcal{L}(i) \in \nu_b(j)$, 0 otherwise and $\mathbf{S}_{l,i}$ is the joint's motion subspace spatial vector defined such that

$\mathbf{v}_l = \mathbf{S}_{l,i} \dot{q}_{l,i} + \mathbf{v}_l$. If position, velocity and acceleration of each link of the rigid body tree are available (e.g., computed by the forward step of the RNEA), then one can calculate the dynamics regressor \mathbf{Y} using (8), (11) and (12).

IV. INERTIAL PARAMETER IDENTIFICATION AND TORQUE ESTIMATION

The rRNEA described in Section III for estimating simultaneously (internal) torques $\boldsymbol{\tau}$ and (external) force \mathbf{f}_β redefines the base link according to the link on which the external force is acting. It is therefore of interest to understand the relationship between ${}_{b,\mathcal{L}}\mathbf{Y}_n$, ${}_{b,\mathcal{L}}\mathbf{Y}_c$ and ${}_{\beta,\mathcal{L}}\mathbf{Y}_n$, ${}_{\beta,\mathcal{L}}\mathbf{Y}_c$ for arbitrary choices of the base link β .

A. Estimation regressors identifiability

In order to simplify the notation we will indicate ${}_{b,\mathcal{L}}\mathbf{Y}_n$ with $\mathbf{Y}_{b,n}$ and similarly ${}_{b,\mathcal{L}}\mathbf{Y}_c$ with $\mathbf{Y}_{b,c}$. In this section we prove the major theoretical result of the present paper. In practice we demonstrate that the inertial parameters used in the procedure in Section III are a subset of the base parameters associated with the regressors associated to the embedded F/T sensors. Let's call this base b and denote the associated identifiable subspace by $I_{\mathbf{Y}_{b,n}}$. The rRNEA in Section III redefines a new base link on the link on which the unknown external wrench is acting, denoted by β . The associated base parameters are therefore identified by $I_{\mathbf{Y}_{\beta,n}}$; similarly, the inertial parameters to compute internal torques are identified by $I_{\mathbf{Y}_{\beta,c}}$. What we will show in Proposition 1 is that: $I_{\mathbf{Y}_{\beta,n}} = I_{\mathbf{Y}_{b,n}}$ and $I_{\mathbf{Y}_{\beta,c}} \subseteq I_{\mathbf{Y}_{b,n}}$. To reach this conclusion we first need another property:

Lemma 1: given two different choices b and β of the base link in a kinematic tree we have:

$$I_{\mathbf{Y}_{b,n}} = I_{\mathbf{Y}_{\beta,n}}. \quad (13)$$

Proof: From (11) it can be easily concluded that: $\mathbf{Y}_{\beta,n} = {}^\beta\mathbf{X}_b^* \mathbf{Y}_{b,n}$ and the (13) easily follows from the observation that the rigid transformation ${}^\beta\mathbf{X}_b^*$ is always full rank and invertible. ■

Proposition 1: given two different choices b and β of the base link in a kinematic tree the induced identifiable subspaces satisfy the following properties:

$$I_{\mathbf{Y}_{\beta,n}} = I_{\mathbf{Y}_{b,n}}, \quad I_{\mathbf{Y}_{\beta,c}} \subseteq I_{\mathbf{Y}_{b,n}}. \quad (14)$$

Proof: For any choice of the base, let us first observe that $\mathbf{Y} = [\mathbf{Y}_n^\top \mathbf{Y}_c^\top]^\top$ implies: $N_{\mathbf{Y}} = N_{\mathbf{Y}_n} \cap N_{\mathbf{Y}_c}$, and therefore: $N_{\mathbf{Y}}^\perp = N_{\mathbf{Y}_n}^\perp + N_{\mathbf{Y}_c}^\perp$, which can be rewritten as $I_{\mathbf{Y}} = I_{\mathbf{Y}_n} + I_{\mathbf{Y}_c}$, using the definition (6) and the usual definition of the operator '+' on vector spaces³. The equation above implies that $I_{\mathbf{Y}_n} \subseteq I_{\mathbf{Y}}$ and $I_{\mathbf{Y}_c} \subseteq I_{\mathbf{Y}}$. Recently it has been shown [2] that $I_{\mathbf{Y}} = I_{\mathbf{Y}_n}$ and therefore we can also conclude $I_{\mathbf{Y}_c} \subseteq I_{\mathbf{Y}_n}$. Now using the Lemma 1 we can consider two different choices b and β of the base link and conclude: $I_{\mathbf{Y}_{b,c}} \subseteq I_{\mathbf{Y}_{b,n}} = I_{\mathbf{Y}_{\beta,n}} \supseteq I_{\mathbf{Y}_{\beta,c}}$. The

³The proof is a simple linear algebra exercise. More in general, it can be proven [12] that vector subspaces with the operators '+', \cap and \perp (the orthogonal complementary space) form an orthocomplemented lattice on which the above property corresponds to the De Morgan law.

last equations clearly include both inclusions in (14) and therefore conclude our proof. ■

Proposition 1 mathematically proves that exactly the same inertial parameters estimated from the F/T sensor measurements can be used to estimate joint torques and external wrenches with the procedure outlined in Section III. In particular, with the F/T sensor measurements we can estimate the inertial parameters in $I_{Y_{b,n}}$ being b the rigid link hosting the F/T sensor. The rRNEA redefines the base at β (contact location) and requires the inertial parameters in $I_{Y_{\beta,c}} \subseteq I_{Y_{b,n}}$ to estimate the joint torques. Similarly it requires the inertial parameters in $I_{Y_{\beta,n}} = I_{Y_{b,n}}$ to estimate the external wrench.

V. EXPERIMENTAL SETUP AND TESTS

In this section we present the results obtained on the iCub humanoid robot. Section V-A presents the software and Section V-B presents the experiments.

A. Software

At the best of our knowledge, no free software library is available for calculating dynamics regressors for floating-base systems. The only two software of which we are aware are Symoro+ [13], which is not freely available⁴, and SymPy-Botics [14], which is limited to chain structures. Anyway, none of them provide the torque estimation regressor used in this work. For this reason we developed the necessary algorithms and added them to the C++ iDynTree library⁵, which is freely available under the GNU GPL license.

B. Validation

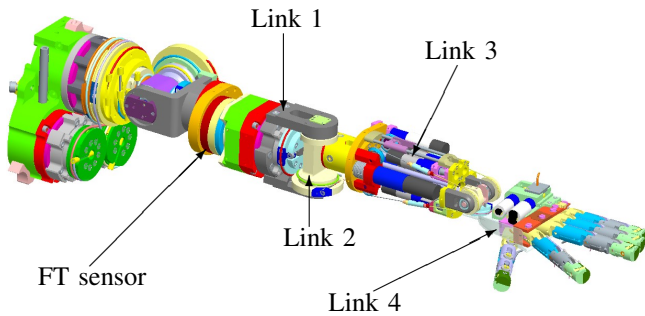


Fig. 2. CAD drawing of the seven degree-of-freedom iCub arm used in the experiments. Three out of the four joints (two in the shoulder and one in the elbow) are sensorized with joint level torque sensors. These joints are the one considered in the proposed experiments.

Experiments were conducted on three joints (pitch and yaw in the shoulder and elbow)⁶ of the iCub left arm (see Fig. 2). These joints are equipped with joint level torque sensors. Additionally a single F/T sensor is positioned in

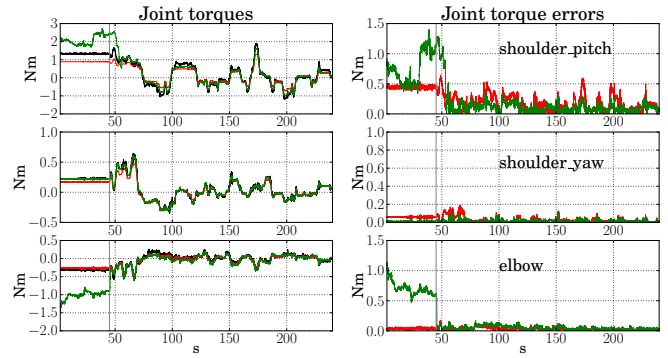


Fig. 3. Joint level torques (left part) and errors (right part): measured (black), estimated with CAD parameters (red) and estimated with the procedure in [1] supplied with identified parameters (green). The vertical solid gray line indicate the movement onset.

the middle of the upper arm as represented in Fig. 2. In the experiment, data from the F/T sensor were used to estimate the associated base parameters. Thanks to the theoretical results presented in the previous sections, these parameters coincides with the one used by the method in the rRNEA to obtain an estimation of the joint torques. These estimations have been compared with direct joint torque measurements, used in this framework as a ground truth. Results are presented in Fig. 3 where we reported in blue direct joint torque measurements, in red predictions using CAD parameters and in green predictions from the estimation in [1] supplied with the on-line estimation of the base parameters (presented in Fig. 4). At the beginning of the simulation estimated parameters are clearly not sufficiently well estimated to predict with sufficient accuracy the joint torques. This condition holds true until the arm starts moving (vertical solid black line in both Fig. 3 and Fig. 4).

During the testing trajectories, the end-effector randomly moved in Cartesian space, without any interaction with the environment. Literature on suitable choices of the exciting trajectories is extensive, but such an implementation is out of the scope of the present paper. This simpler choice is also motivated by two factors: first, we need to avoid self-collision of the robot in a simple way; second, we need to generate trajectories similar to those produced during standard operation of the humanoid robot and our goal is to on-line estimate the parameters during standard operations. Joint velocities and accelerations have been estimated using an adaptive-window fitting algorithm [15].

VI. CONCLUSION

In this paper we presented some theoretical and numerical advances with respect to the problem of estimating joint torques from proximal force and torque (F/T) sensors. This estimation problem was originally proposed in [1] and it is based on the idea of exploiting the simple Newton-Euler recursion step to propagate force and torque information across a kinematic structure. The estimation relies on the knowledge of the system inertial parameters. In this paper we address the problem of estimating these parameters directly

⁴There is however a recent partial porting on python: <https://github.com/symoro/symoro>

⁵<http://wiki.icub.org/wiki/iDynTree>

⁶<http://wiki.icub.org/wiki/ICub.joints>

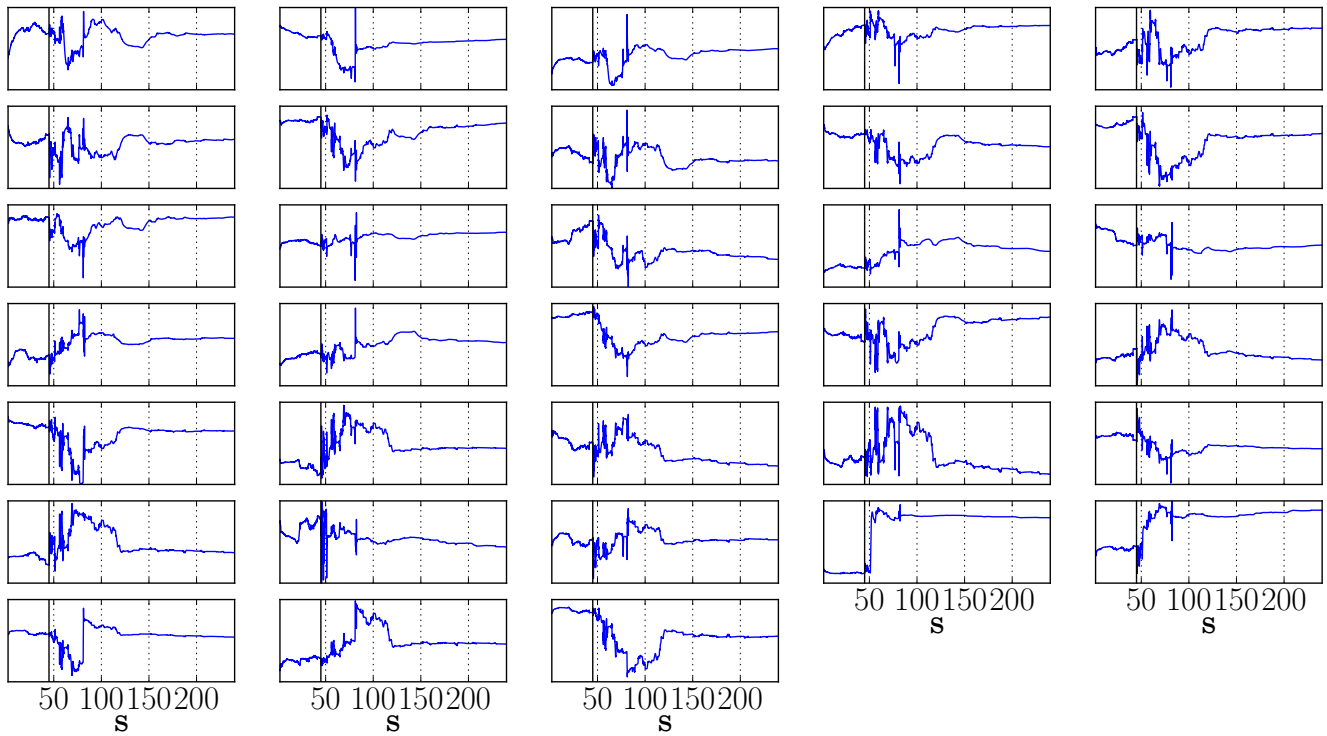


Fig. 4. The picture shows the time behavior of the base parameters estimation. The estimation is executed on-line in an iterative fashion. The onset of the movement (vertical solid black line) determines the instant at which the data from the F/T sensor become informative for the estimation problem. Convergence is quite fast and mirrors the behavior of the torques estimation in Fig. 3.

from the F/T sensor. It is in particular shown that the parameters identifiable from the F/T sensor (the so called base parameters) coincide with those used by the joint torque estimation procedure. This result has been obtained by extending some previous findings [2].

Validation of the proposed theoretical framework has been conducted on the iCub humanoid robot, which is equipped with both F/T sensors and joint torque sensing. On-line estimation of the base inertial parameters have been performed by means of F/T sensor measurements only. The estimated base parameters have been used to compute joint torques from F/T measurements as in [1]. Comparison with direct joint torque measurement (used in this context as ground truth) shows the efficacy of the proposed estimation procedure.

REFERENCES

- [1] M. Fumagalli, S. Ivaldi, M. Randazzo, L. Natale, G. Metta, G. Sandini, and F. Nori, "Force feedback exploiting tactile and proximal force/torque sensing. Theory and implementation on the humanoid robot iCub," *Autonomous Robots*, vol. 33, no. 4, pp. 381–398, 2012.
- [2] K. Ayusawa, G. Venture, and Y. Nakamura, "Identifiability and identification of inertial parameters using the underactuated base-link dynamics for legged multibody systems," *The International Journal of Robotics Research*, 2013.
- [3] J. Hollerbach, W. Khalil, and M. Gautier, "Model Identification," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer Berlin Heidelberg, 2008, pp. 321–344.
- [4] B. S. R. Armstrong, "Dynamics for Robot Control: Friction Modeling and Ensuring Excitation During Parameter Identification," Ph.D. dissertation, Stanford University, 1988.
- [5] D. Nguyen-Tuong and J. Peters, "Using model knowledge for learning inverse dynamics," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 2677–2682.
- [6] M. Fumagalli, A. Gijsberts, S. Ivaldi, L. Jamone, G. Metta, L. Natale, F. Nori, and G. Sandini, *From Motor to Interaction Learning in robotics*. Springer-verlag, 2010, ch. Learning how to exploit proximal force sensing: a comparison approach, pp. 159–177.
- [7] R. Featherstone, "A Beginner's Guide to 6-D Vectors (Part 1)," *Robotics Automation Magazine, IEEE*, vol. 17, no. 3, pp. 83–94, sept. 2010.
- [8] —, *Rigid body dynamics algorithms*. Springer Berlin., 2008, vol. 49.
- [9] H. V. Henderson and S. R. Searle, "Vec and vech operators for matrices, with some uses in jacobians and multivariate statistics," *Canadian Journal of Statistics*, vol. 7, no. 1, pp. 65–81, 1979.
- [10] T. Iwasaki, G. Venture, and E. Yoshida, "Identification of the inertial parameters of a humanoid robot using grounded sole link," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2012, pp. 449–454.
- [11] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *Int. J. Rob. Res.*, vol. 5, no. 3, pp. 101–119, Sept. 1986.
- [12] G. Birkhoff, *Lattice theory*. AMS Bookstore, 1967, vol. 25.
- [13] W. Khalil, D. Creusot, et al., "Symoro+: a system for the symbolic modelling of robots," *Robotica*, vol. 15, pp. 153–161, 1997.
- [14] C. D. Sousa and R. Cortesão, "Sagerobotics: open source framework for symbolic computation of robot models," in *Proc. of the 27th Annual ACM Symposium on Applied Computing*, 2012, pp. 262–267.
- [15] F. Janabi-Sharifi, V. Hayward, and C.-S. Chen, "Discrete-time adaptive windowing for velocity estimation," *Control Systems Technology, IEEE Transactions on*, vol. 8, no. 6, pp. 1003–1009, nov 2000.